BlackBerry Java Application

Version: 5.0 Beta

Release Notes

BlackBerry®

Contents

Product information

<div style="float:right">1</div>

The BlackBerry® Java® Development Environment is a fully integrated development environment and simulation tool for building a BlackBerry Java Application for a BlackBerry device.

The BlackBerry JDE is a Mobile Information Device Profile (MIDP) compliant with Java® ME environment for developers.

Use the BlackBerry JDE to build applications that take advantage of the unique features of the BlackBerry device, such as desktop synchronization applications, GPS applications, and applications that push content proactively to BlackBerry devices in environments that use the BlackBerry® Enterprise Server. The BlackBerry JDE provides ways for BlackBerry device applications to establish network connections to servers on the Internet or the corporate intranet.

Each version of the BlackBerry JDE comes with the BlackBerrySmartphone Simulators that were available when that version of the BlackBerry JDE was made public. To download additional BlackBerry Smartphone Simulators, visit www.blackberry.com/developers/.

Internationalization

The BlackBerry JDE contains tools that assist with internationalization (i18n) efforts. It is important to follow naming conventions. The Localization Demo sample application can be reviewed for more information because it makes use of i18n.

Note: The .rrc and .rrh files must exist in the same directory for successful compilation and you should define resource keys in uppercase letters only.

For more information about localization, see the *BlackBerry Java Application Development Guide* or the *BlackBerry Integrated Development Environment Online Help*.

Code samples

A sample BlackBerry JDE workspace exists in the samples subdirectory of the BlackBerry JDE installation directory. It contains sample programs that demonstrate how to design and build Java applications for BlackBerry devices.

Code samples with server components

The following sample applications include server components. Invoke the `run.bat` script in the specified folder to start the server component:

- The Bluetooth® Serial Port Demo sample includes a server component at: samples/com/rim/samples/server/bluetoothserialportdemo
- The Browser Multipart Push Demo sample includes a server component at: samples/com/rim/samples/server/browsermultipartpushdemo
- The Browser Push Demo sample includes a server component at: samples/com/rim/samples/server/browserpushdemo
- The GPS Demo sample includes a server component at: samples/com/rim/samples/server/gpsdemo
- The HTTP Push Demo sample includes a server component at: samples/com/rim/samples/server/httppushdemo
- The SMS Demo sample includes a server component at: samples/com/rim/samples/server/smsdemo

- The Socket Demo sample includes a server component at: samples/com/rim/samples/server/socketdemo

System requirements

Item	Requirement
computer	- Intel® Pentium® processor or compatible (2 GHz minimum) - minimum 1 GB of RAM or more available - minimum 500 MB of disk space available
operating system	Any of the following operating systems: - Windows Vista™ (32-bit) - Windows® XP Professional (32-bit)
Java® SE Development Kit	Java SE Development Kit version 1.6 or later (32-bit)
Microsoft® DirectX®	Microsoft DirectX version 8.0 or later

New in this beta release

This section lists functional changes for the BlackBerry® Java® Development Environment version 5.0.0 build 32.

User interface

Item	Description
Auto complete field	The `net.rim.device.api.ui.component.AutoCompleteField` is a UI component that automatically matches what the user types in the field against items in a datasource and displays the matches in a droplist below the field. The items in the datasource are defined by the `BasicFilteredList` class and can be a static or dynamic list (i.e. the Contact list).

New in this release

This section lists functional changes for the BlackBerry® Java® Development Environment version 5.0.0.

Application integration

Item	Description
Enhanced BrowserField	You can embed web content within a BlackBerry device application using the `net.rim.device.api.browser.field2.BrowserField` class. This field takes up the dimensions of web content rendered within it. Several methods are provided to register listeners, request, execute, scale, display, debug and navigate content, retrieve content information, and handle errors.
Widgets API	You can now register third-party JavaScript® extensions with a widget using the `net.rim.device.api.web.WidgetExtension` interface that is implemented by the `net.rim.device.api.web.jse.BlackBerryWidgetToolkit` class. This is used by the Web Component Pack to add extensions to the Script Engine.
Widget Extension API's	You can now use the `net.rim.device.api.script` package to develop custom BlackBerry Widget extensions. You can build a library using the following interfaces and classes to extend scripting capabilities: • `Scriptable` • `ScriptableFunction` • `ScriptableImpl` • `ScriptExtension`
MEID Retrieve API	You can now retrieve the MEID from a BlackBerry device that operates on a CDMA network. You can retrieve a string representing the decimal or hex MEID using methods available from the `net.rim.device.api.system.CDMAInfo` class.
Multi-line API	You can now access various information about the multiple lines (for example, a cellular line and work line) available on BlackBerry devices. You can retrieve the phone numbers of the available lines, determine which line to use, and initiate outgoing calls with selected phone numbers using new methods available form the `net.rim.blackberry.api.phone` package. The new Multi-line API provides the following features: • retrieve the phone numbers and labels, IDs, and WAFs of available lines

Item	Description
	• determine whether a phone line is a cellular line or a work line • invoke the BlackBerry phone application using a source line and destination number The `MultiLineListener` abstract class includes all callbacks from the `PhoneListener` interface.
PhoneScreenField replacement	You can use the new `net.rim.blackberry.api.phone.phonegui.PhoneScreen()` method to perform actions such as add, delete, replace field, etc. The `PhoneScreenField()` has been removed. `PhoneScreenVerticalManager()` organizes field objects to be displayed vertically and `PhoneScreenHorizontalManager` organizes field objects to be displayed horizontally. Further enhancements have been made to the `net.rim.blackberry.api.phone.Phone` and `net.rim.blackberry.api.phone.MultiLineListener` classes. For more information on updating the phone screen, see the *BlackBerry Java Application Integration Development Guide*.
MIDP 2.1 support	MIDP 2.1 is supported in this release.
Content-Folder support	You can specify the folder on the BlackBerry device where you want to place your BlackBerry device application by adding a `Content-Folder` attribute to your application's .jad file. For example, to place your application in the Games folder, add `Content-Folder: Games` to the .jad file.
Improvements to WBXMLWriter and WBXMLParser	`net.rim.device.api.xml.jaxp.WBXMLWriter` and `WBXMLParser` include new methods to ease the provisioning of three tables, and allow for support of multiple codepages. The following methods have been added to the `WBXMLWriter` class: • `void setAttrStartTable(String[] attrStartTable)` • `void setAttrValueTable(String[] attrValueTable)` • `void setTagTable(String[] tagTable)` The following methods have been added to the `WBXMLParser` class: • `Vector getNameSpaceMap()` • `void setAttributePageFollowTag(boolean follow)` • `void setAttrStartTable(String[] attrStartTable)`

Item	Description
	• `void setAttrValueTable(String[] attrValueTable)` • `void setTagTable(String[] tagTable)`

Core I/O connection framework

Item	Description
Biometric API	You can use the Biometric API to extract fingerprint information from a smart card to match against a scanned fingerprint. Extensions have been made to the Smart Card API to allow other smart card manufacturers to return fingerprint information. You can store, manage, and retrieve fingerprint data using functionality available from the `net.rim.device.api.biometrics` package. You can determine if fingerprint authentication is supported and retrieve the fingerprint information using new methods available from the `net.rim.device.api.smartcard.SmartCard` and `net.rim.device.api.smartcard.SmartCardSession` classes.
Network API	The Network API attempts to simplify the establishment of connections over the different transports available on the BlackBerry device. The `net.rim.device.api.io.transport` and `net.rim.device.api.io.transport.options` packages contain the implementation of the Network API. It provides the means for querying the availability of transports and for selecting the most appropriate one to establish a connection. It abstracts the details of parsing `ServiceRecords` and constructing proper URLs for `Connector.open()`. The Network API is intended for use with http://, https://, socket://, tls://, and ssl:// connections.
User Authentication API	You can employ a separate token (such as a smart card or a proximity token) for user authentication on a BlackBerry device using the User Authentication API available from the `net.rim.device.api.userauthenticator` package. • The `PasswordFrameworkProvider()` interface allows for the implementation of user authenticator to provide the required information to participate in the password-unlock process. • The `SecurityUIManager` class provides simple user-interface services to user authenticators. • The `UserAuthenticator` class provides the primary means of interaction with a user authentication token.

Item	Description
	• The `UserAuthenticatorContextObject` class is used to create context objects that are employed to pass generic data between a user authenticator and the security framework. • The `UserAuthenticatorManager` class is a system-wide singleton manager that is used to handle user authenticators loaded on the BlackBerry device.
Hotspot API	The classes and interfaces provided in `net.rim.device.api.wlan.hotspot` can be used to implement your own Hotspot Client that users can use to connect to your Wi-Fi® hotspots.
Push API	You can now use a new Push API in the `net.rim.blackberry.api.push` package to access the following functionality: • develop scalable push applications • store registration information of subscriptions to push services • automatically start a push application when a push message arrives and optionally bring the application to the foreground • register with a BlackBerry Push API server • detect SIM card swaps and automatically unregister SIM cards • automatically unregister from a BlackBerry Push API server when an application is removed from the BlackBerry device

Data storage

Item	Description
Random File Access API	You can read data from and write data to a file at any location within the file using the Random File Access API. A new `net.rim.device.api.io.Seekable` interface is provided in the `net.rim.device.api.io` package. Two new methods introduced in this interface enable you to get and set the current position within a stream.
SQLite® Support	You can now develop applications that use the integrated SQLite database using the SQLite API. You can use the following classes in the `net.rim.device.api.database` to work with a SQLite database: • `DatabaseFactory` is a factory class you can use to get database access by opening an existing database or creating a new one. • `Database` provides a means to create statements.

Item	Description
	• `Statement` features values binding after it has been prepared. You can use the `execute()` method for statements such as `INSERT`, `DELETE`, `UPDATE`. You can use the `getCursor()` method for query statement such as `SELECT`.
	• `Cursor` provides iterative operations on the row set.
	• `Row` provides data retrieval operations.
	• `BufferedCursor` provides a means to cache data and access rows in random or bidirectional manners.
	• `CursorEnumeration` provides a simplified method of cursor navigation.
	For more information on the SQLite API, see the *BlackBerry Java Application SQLite Development Guide.*

Multimedia

Item	Description
Bitmap Scale and Compare API	You can scale and compare bitmaps using the Bitmap Scale and Compare API. Methods in the `net.rim.device.api.system.Bitmap` class were updated and new APIs were added. The updated methods provide functionality that include scaling of bitmaps, getting scaled bitmap information, setting available filter types, and comparing bitmaps.
Enhanced Media Key Events	BlackBerry device applications can now respond to key presses of media keys even when the BlackBerry device application is in the background. This allows a BlackBerry device user to control media playback using certain keys even while another BlackBerry device application is in the foreground. This is done by implementing the new interface, `net.rim.device.api.media.MediaActionHandler`, which defines a single method `boolean mediaAction(int action, int source, Object context)`. The implementation is registered with the new method, `Application.addMediaActionHandler(MediaActionHandler handler)`.
	After registration, the `MediaActionHandler`'s method will be notified when certain keys are pressed that correspond to media "actions", such as "volume up", "volume down", "next track", "previous track", and "play/pause toggle". These notifications continue even when the BlackBerry device application is in the background. For handling media key presses while the BlackBerry device application is in the foreground, the new abstract class `net.rim.device.api.media.MediaKeyListener` can be used.

Item	Description
	All registered `MediaActionHandler` instances are notified of media key presses on connected wired headsets and paired Bluetooth® headsets. When media actions come from these sources the value of the `source` parameter specified to `mediaAction()` is `SOURCE_WIRED_HEADSET` or `SOURCE_BLUETOOTH_HEADSET`, respectively. Also, a non-null value will be specified for `context` when actions come from these headset sources. This feature is not available in BlackBerry JDE version 5.0.0 build 20.
Video capture	You can now embed a viewfinder, record video clips, and play captured video using the functionality specified in JSR 135. When your BlackBerry device application records to a file, the `Player` records in 3GP format. When your application records to an to an OutputStream, the `Player` records in a proprietary streaming video format. You can retrieve and control a video viewfinder, and record the video shown in the viewfinder using the `VideoControl` and `RecordControl` controls. The `Player` records to a file specified by a file locator or to an OutputStream. A new `Player` must be created to display the recorded video. Controls used to get the viewfinder and record video can be retrieved from a `Player` created by `Manager.createPlayer("capture://video?encoding=video/3gpp");`.
Advanced Multimedia Supplements	The Advanced Multimedia Supplements of JSR 234 are now supported and accessed through `javax.microedition.amms`. The Advanced Multimedia Supplements add many new controls for advanced camera and video features.
Image API	The `net.rim.device.api.ui.image.Image` class represents an image of agnostic type that can be painted into a graphics context. This is recommended for containers or components that support and display different types of images such as Bitmap, BMPEncodedImage, PNGEncodedImage, etc. An image cannot be instantiated directly. Different types of images can be created by calling methods in the `net.rim.device.api.ui.image.ImageFactory` class.

PIM (Contacts, Calendar, Tasks, MemoPad)

Item	Description
Multiple ContactList support	Multiple ContactList support provides individual access to each ContactList. The following methods in `javax.microedition.pim.PIM` have been modified to query and open PIM lists:

Item	Description
	• `public String[] listPIMLists(int pimListType)` • `public PIMList openPIMList(int pimListType, int mode)` • `public PIMList openPIMList(int pimListType, int mode, String name)`
Name reading support in contacts	You can now access the Japanese reading of the `NAME_GIVEN` and `NAME_FAMILY` elements of the `NAME` and `ORG` fields using the `net.rim.blackberry.api.pdap.BlackBerryContact` interface. You can access this information using the `NAME_GIVEN_YOMI` and `NAME_FAMILY_YOMI` indices of the `NAME String` array field's value, and the new `ORG_YOMI` field.
PIMList UID	You can now reference a PIMList by its UID. It may be preferable to reference a PIMList by its UID instead of its name because a UID is more constant. The name of a PIMList can change at any time and a different name may be assigned to the same PIMList after a software upgrade. In contrast, a PIMList's UID does not change during the lifetime of the PIMList and remains the same after the BlackBerry device software is upgraded. BlackBerry device applications that maintain a parallel database will want to reference the PIMList by its UID instead of its name in order to maintain referential integrity. The `getPIMListUID()` method in `net.rim.blackberry.api.pdap.BlackBerryPIMList` has been added to return the unique identifier for this PIMList. The `openPIMList(int pimListType, int mode, long uid)` method in `net.rim.blackberry.api.pdap.BlackBerryPIM` has been added to open a PIMList by its UID.
Unified PIMLists	You can use the following `net.rim.blackberry.api.pdap.BlackBerryPIM.openUnifiedPIMList()` methods to view of two or more PIMLists with their items merged together: • BlackBerryPIMList openUnifiedPIMList(int pimListType, int mode) • BlackBerryPIMList openUnifiedPIMList(int pimListType, int mode, String[] names) • BlackBerryPIMList openUnifiedPIMList(int pimListType, int mode, long[] uids)
Additional BlackBerry PIMList methods	The following methods of the `net.rim.blackberry.api.pdap.BlackBerryPIMList` interface are now available:

Item	Description
	• `public boolean isWirelessSyncCapable()`
	• `public boolean isWirelessSyncEnabled()`
	• `public int size()`
Linking application contacts to BlackBerryContacts	The `net.rim.blackberry.api.pdap.contactlinking` package provides functionality to link application contacts to `BlackBerryContacts`.
	The following classes are available for use:
	• `LinkedContactUtilities`
	• `AbstractLinkableContact`
	• `DefaultLinkableContact`
	The following interfaces are available for use:
	• `AddressBookFieldFactory`
	• `LinkableContact`
	• `LinkedContactConstants.`

Messaging

Item	Description
Attachment API	You can programmatically download attachments to email messages using the `net.rim.blackberry.api.mail.AttachmentDownloadManager` class.

User interface

Item	Description
Absolute layout manager	The `net.rim.device.api.ui.container.AbsoluteFieldManager` class provides a `Manager` that allows placement of fields at specific x-y positions. Rather than based on the extent of the surrounding fields, fields can be placed anywhere on the screen and can even be overlapped.
Auto complete field	The `net.rim.device.api.ui.component.AutoCompleteField` is a UI component that automatically matches what the user types in the field against items in a datasource and displays the matches in a droplist below the field. The items in the datasource are defined by the `BasicFilteredList` class and can be a static or dynamic list (i.e. the Contact list).

Item	Description
Date and time picker	The `net.rim.device.api.ui.picker.DateTimePicker` class is a user interface component that is used to only pick a date, only pick a time, or pick the date and time together. While the `DateTimePicker` is a `Manager`, no additional fields are actually intended to be added to this.
Enhancements for string comparisons	The `compareToHandleSpecialChars(String s1, String s2)` method in the `net.rim.device.api.util.StringUtilities` class can be used to do a case-insensitive comparison of two strings that handles special characters differently than strict ASCII order. This method replaces `StringUtilities.compareToOutlookOrder(String, String)`.
EventInjector API	The `net.rim.device.api.system.EventInjector` class has been improved for BlackBerry devices with a touch screen to allow for additional programmatic injection of events into the system using the `EventInjector` class. The `EventInjector.TouchEvent` has been added which includes fields and methods to set and get points with X and Y coordinates, as well as create and inject touch gestures such as swipe, tap, and click through.
Eyelid field	The `net.rim.device.api.ui.extension.container.EyelidFieldManager` class is used to display text or static image information on the top of a UI object for a period of time.
FontManager API	You can now load and remove custom fonts for system-wide or application-specific use using the `net.rim.device.api.ui.FontManager` class. You can load custom fonts into persistent memory (see `SYSTEM_FONT`) or into application memory (see `APPLICATION_FONT`).
Menu Icon API	You can add icons to a menu using new methods available from the `net.rim.device.api.ui.MenuItem` class. You can set a menu icon and determine an existing icon by using the `setIcon(Image icon)` and `Image getIcon()` methods.
Menu Customization API	The `net.rim.device.api.ui.component.Menu` class has been enhanced to allow you to determine and set the background, border and font for the menu. When displayed, the menu will be shown with the custom background, border and font. The following methods have been added to enable this functionality: • `getBackground()` • `setBackground(Background)` • `getBorder()`

Item	Description
	• `setBorder(Border)` • `getFont()` • `setFont(Font)`
Picture scrolling	The `net.rim.device.api.ui.extension.component.PictureScrollField` class is used to draw a row of images that can be scrolled from side-to-side using trackball or touch gestures.
Screen Transitions API	The `net.rim.device.api.ui.TransitionContext` class contains all the necessary data to uniquely describe a transition animation between two screens. An instance of this class represents one of the following five types of standard animations: • TRANSITION_FADE • TRANSITION_SLIDE • TRANSITION_WIPE • TRANSITION_ZOOM • TRANSITION_NONE Each animation type supports a set of attributes that allow its behavior to be customized.
Spin box manager	The `net.rim.device.api.ui.container.SpinBoxFieldManager` class is used to manage a set of `SpinBoxFields`. The spin box manager dictates the height of each row and the number of rows that are shown for each spin box. If a `SpinBoxField` is owned by a `Manager` then `SpinBoxField.setVisibleRows` and `SpinBoxField.setRowHeight` will have no effect. Note that row heights are fixed. Each row must have the same height.
Zoom screen	The `net.rim.device.api.ui.extension.container.ZoomScreen` class provides a capability to zoom-in and pan around images using API, with multi-touch used to define the zoom area.

GPS & Location-based services

Item	Description
JSR 179 Extensions	The `net.rim.device.api.gps` package has been enhanced to include the following extensions to JSR 179: • GPSInfo()

Item	Description
	• BlackBerryLocationProvider() • BlackBerryCriteria() • BlackBerryLocation() • SatelliteInfo()
Reverse Geo-Coding	You can now use latitude and longitude points to determine an address, city, province/state, or country using new methods available from the `net.rim.device.api.lbs.Locator` class. The class provides `Landmark[] reverseGeocode(Coordinates coord, int searchType)` and `Landmark[] reverseGeocode(int latitude, int longitude, int searchType)`.
Location Picker API	The Location Picker API, contained in the `net.rim.device.api.lbs.picker` package, permits applications to invoke the `LocationPicker` to display a graphical list for a user to select a location from. The list of facilities is configurable and might include the following: • "My current location…" (GPS): If the BlackBerry device supports GPS, or if cell tower triangulation is available, then the user's current location may be selected. • "Find on a map…" (LBS): A user can search, pan, rotate and zoom the map to select a location. Developers can mark predefined locations on the map. If GPS is supported then the user's current location may also be shown on the map. The user will have access to their favorites and recent locations within the LBS application. • "Recently selected…" (Previously selected locations): The `LocationPicker` stores 10 recent selections by a user, which may be used by all applications. • "Application suggestions…" (predefined locations): You may provide predefined locations in the list. • "Using Google Maps™" (Registered location providers): Applications such as Google Mapscan register as location providers to provide alternative resources for users to select a location. • "Enter a location…" (Manual entry): A user may enter an address / location via the keyboard.

Debugging and refining code

Item	Description
On-Device Profiling	The BlackBerry JDE has been improved to retrieve profiling data from a BlackBerry device that has been attached to the BlackBerry JDE.

Fixed issues

This section lists fixed issues for the BlackBerry® Java® Development Environment version 5.0.0 build 32.

Item	Description
225857	If you attempted to compile a class that extends either `net.rim.device.api.wlan.hotspot.HotspotAuthenticationAgent` or `net.rim.device.api.wlan.hotspot.HotspotCredentialsAgent`, a compilation error occurred.
215913	The `net.rim.device.api.wlan.hotspot.HotspotInfo` class was not visible in the API reference for the BlackBerry JDE.
SDR367368	The `BrowserField2` API was not included in the BlackBerry JDE.
SDR361470	The `ImageFactory.createImage(EncodedImage encodedImage)` method in the `net.rim.device.api.ui.image` package was not included in the BlackBerry JDE.
SDR359947	The `net.rim.device.api.ui.TransitionContext` class resulted in a message that a RIMAPPSA2 signature was required.
SDR359070	BlackBerry device applications that create an unencrypted SQLite® database did not work when run on a BlackBerry Smartphone Simulator in insecure mode.
SDR336115	Invoking new `net.rim.device.api.database.DatabaseSecurityOptions(int signerId)` with a valid `signerid` within a signed .cod file threw an `IllegalArgumentException` even if the .cod file was signed with that `signerid`.
SDR336034	If you used the BlackBerry JDE, debug information was missing from several modules.
SDR335767	Clicking a `net.rim.device.api.browser.field2.BrowserField` URL link triggered an exception thrown by the `net.rim.device.api.browser.field2.ProtocolController` in certain circumstances.
SDR332438	Adding a `net.rim.device.api.ui.component.EditField` to a `net.rim.blackberry.api.phone.phonegui.PhoneScreenField`, and then invoking `PhoneScreenField.sendDataToScreen()` caused the BlackBerry Smartphone Simulator to crash.
SDR329815	The following public APIs resulted in a message that a RIMAPPSA2 signature was required: • `net.rim.device.api.browser.field2.*`

Item	Description
	• `net.rim.device.api.browser.field2.debug.*` • `net.rim.device.api.script.*` • `org.w3c.dom.html2.*` • `net.rim.device.api.io.Seekable`
SDR328588	`EventInjector.TouchEvent.setPoint2(int x1, int y1)` threw an `IllegalArgumentException` when invoked with valid coordinates, the first touch point was valid, and the event was MOVE.
SDR306049	BlackBerry device applications using `StringUtilities.compareToOutlookOrder(String s1, String s2)` did not compile. Existing BlackBerry device applications that used this method threw a run-time exception error on BlackBerry devices that were running BlackBerry® Device Software version 5.0.0.
SDR305298	Debug files were reported as "Not Found" when you attached a BlackBerry device to the BlackBerry JDE in a specific sequence.
SDR305018	After loading BlackBerry device applications onto a BlackBerry device using JavaLoader, the BlackBerry device applications did not appear in the Download folder of the BlackBerry device until the BlackBerry device was reset.
SDR304948	If you used BlackBerry JDE to compile and then access BlackBerry device applications using the BlackBerry Smartphone Simulator, opening the code profiling window by selecting View > Profile caused the BlackBerry JDE to immediately terminate while leaving the BlackBerry Smartphone Simulator running.
SDR304209	If you used BlackBerry JDE, the default properties window of a multi-value resource key was too small to show all keys.
SDR304006	If you used BlackBerry JDE, the Home Screen Position field under project properties did not affect the order of BlackBerry device applications.
SDR300638	If you used the BlackBerry Smartphone Simulator, you might have encountered a JVM Error 104, array index out of bounds exception.
SDR215764	If you used the BlackBerry Email Simulator, the simulator did not send or receive email messages.
SDR215580	If you used the `updatejad` tool, included in the BlackBerry JDE, to update duplicate .jad files, the `updatejad` tool did not display a warning message and incorrectly updated the .jad file.
SDR150080	If you use the BlackBerry JDE, a .cod file was not be created in the project directory or the working simulator directory when the Output File Name was set to blank ("").
BrowserField API	The `BrowserField` class in `net.rim.device.api.browser.field2` was not fully functional.

Sample applications

Item	Description
217730	If you compiled the BufferedPlayback sample application by using the BlackBerry JDE version 4.6.1 or 4.7.0 and ran the sample application on a BlackBerry device running BlackBerry® Device Software version 5.0.0.168 Bundle 239, the sample application threw a `ClassDefNotFound` exception.
216069	The SVG Midlet Demo application threw an `IllegalArgumentException` when it was launched.
SDR362570	After you compiled and launched the Notifications Demo application in a simulator and selected Notify (ID1) from the menu, the sample application displayed an error.
SDR359045	After you launched the ObjectGroupingDemo sample application, it threw an unexpected exception.
SDR330574	After using the CameraDemo to take a photograph and then attempted to return to the Camera screen, the viewfinder failed to restart.
SDR306751	If you used BlackBerry JDE to compile and then access the sample applications using the BlackBerry Smartphone Simulator, some sample applications did not appear on the Home screen of the BlackBerry device and some BlackBerry device applications could not be opened or threw an exception.
SDR304879	If you used BlackBerry JDE to compile and then access the sample BlackBerry device applications using the BlackBerry Smartphone Simulator, the OTA Sync Demo provided editable text fields to edit a contact but did not save any changes made.
SDR304642	If you used BlackBerry JDE to compile and then access the sample BlackBerry device applications using the BlackBerry Smartphone Simulator, running the SVG Animator Demo threw an uncaught exception (index 2 >= 2).
SDR304238	If you used BlackBerry JDE to compile and then access the sample BlackBerry device applications using the BlackBerry Smartphone Simulator, the Socket Demo application did not print the message, "Received: Hello Received: GoodbyeAndFarewell Done!", to the screen.
SDR304005	If you used BlackBerry JDE to compile and then access the sample BlackBerry device applications using the BlackBerry Smartphone Simulator, and if you used the HTTP Demo and issued two consecutive 'Fetch' commands, the "An outstanding fetch request hasn't yet completed!" dialog box message did not display.
SDR303665	If you used BlackBerry JDE to compile and then access the sample applications using the BlackBerry Smartphone Simulator, the sample Tic Tac Toe application did not highlight the focused square.

Known issues

This section lists known issues for the BlackBerry® Java® Development Environment version 5.0.0 build 32.

Item	Description
351644	Constants in the class `BrowserFieldDebugger` are not accessible but appear in the autocomplete box in the BlackBerry JDE. **Impact:** You cannot use the constants in the `BrowserFieldDebugger` class. **Workaround:** None
347528	You cannot import or use the `net.rim.device.api.ui.menu.SubMenu` class. If you try to import the class, a build failure occurs with the message "package net.rim.device.api.ui.menu does not exist". **Impact:** You cannot use the `SubMenu` class. **Workaround:** None
346804	Adding many records to a database table consecutively can cause the BlackBerry Smartphone Simulator to crash. **Impact:** The BlackBerry Smartphone Simulator crashes. **Workaround:** Do not add records to a database table consecutively.
343875 273122	If you try to sign an application that uses some APIs, including `TransportInfo` and `AutoCompleteField`, the result is a message that a RIMAPPSA2 signature is required. The signing then fails. **Impact:** You may not be able to run your application. **Workaround:** Ignore the message and the application may run as expected. If not, run the BlackBerry Smartphone Simulator in non-secure (default) mode.
339752	If you simulate inserting and formatting an media card using the default directory value, critical files are deleted making the BlackBerry JDE unusable. **Impact:** You cannot use the BlackBerry JDE after attempting to insert and format the media Card. **Workaround:** Do not use the default directory for SD card storage.
339742	If you simulate an incoming call using the BlackBerry Smartphone Simulator, the result is an internal error due to division by zero.

Item	Description
	Impact: You cannot simulate an incoming call using the BlackBerry Smartphone Simulator.
	Workaround: None
339344	When you start multiple instances of the BlackBerry Smartphone Simulator, a `ControlledAccessException` is thrown.
	Impact: You cannot start multiple instances of the BlackBerry Smartphone Simulator.
330676	If you invoke the `Phone.setPreferredLine()` method on a CDMA BlackBerry device, the result is a `RuntimeException`.
	Impact: You cannot invoke the `Phone.setPreferredLine()` method on a CDMA BlackBerry device.
	Workaround: None
324188	If you use the `AutoCompleteField` UI component to search for a contact that has an associated photo, birthday, or anniversary, the result is an exception from the `ContectImpl.equals()` method call during the search.
	Impact: You cannot search for contacts that have an associated photo, birthday, or anniversary field using the `AutoCompleteField`.
	Workaround: None
322255	The drop-down list that appears when using an `AutoCompleteField` UI component remains when the screen containing the `AutoCompleteField` is closed.
	Impact: The screen displayed after the screen containing the `AutoCompleteField` may be obscured.
	Workaround: None
322123	The `BasicFilteredList.removeDataSet()` method does not work.
	Impact: You cannot use the `BasicFilteredList.removeDataSet()` method.
	Workaround: None
273122	You cannot use the Network API on a BlackBerry device.
	Impact: You cannot deploy an application on a BlackBerry device that uses the functionality available from the `net.rim.device.api.io.transport` package.
	Workaround: Test your BlackBerry device application on a BlackBerry Smartphone Simulator.

Item	Description
220277	If you attempt to code sign .cod files using the BlackBerry JDE, a dialog box appears prompting you to generate a private signing key pair. After generating a private signing key, a code signing window appears. When you close the window, the sigtool.csk and sigtool.db files disappear. The application prompts you with the same dialog boxes the next time you attempt to code sign.
	Impact: The BlackBerry JDE private key pair is deleted once you close the signing tool window, with the key pair regenerated by the user the next time they attempt to use the signing tool.
	Workaround: None
SDR368476	If you add a `net.rim.blackberry.api.phone.PhoneListener` within your application and do not remove the listener before the application exits, the application continues to run in the background, and you cannot restart your application. An error message is generated stating that your application is still active.
	Impact: After closing your application, it continues to run in the background.
	Workaround: Invoke `Phone.removePhoneListener(PhoneListener)` to remove active phone listeners before your application terminates.
SDR358926	The icon associated with a BlackBerry device application does not switch to its focus icon when the application is brought into focus.
	Impact: BlackBerry device users do not see application focus icons.
	Workaround: None
SDR336322	Cookies are not set when accessing a web page that sets cookies using a `net.rim.device.api.browser.field2.BrowserField(BrowserFieldConfig)` with the default `net.rim.device.api.browser.field2.BrowserFieldConfig`.
	Impact: Web pages that rely on cookies might not work correctly.
	Workaround: Do not use the default `net.rim.device.api.browser.field2.BrowserFieldConfig`.
SDR324139	Recording, stopping, and committing a video recording, changing the video recording location from File to Stream, and then starting video recording again might cause the BlackBerry device to crash.
	Impact: You cannot change the storage location to stream during a video recording session.
	Workaround: Set the video recording location correctly and do not change it during a recording session.

Item	Description
SDR324117	Video recording to a stream is not working correctly. For example, if you start the Video Recording Demo with the BlackBerry Smartphone Simulator, select any encoding scheme, select Stream from the Recording Location drop-down list, and start recording, the camera display disappears. After stopping and then committing the recording, the sample application throws a `java.lang.IOException` (Camera is busy). **Impact:** You cannot record video to a stream. **Workaround:** None
SDR318654	`net.rim.device.api.browser.field2.BrowserField` does not release focus on a BlackBerry device where the cursor becomes locked in web browsing mode and focus does not leave the field. **Impact:** Focus cannot be away from the `BrowserField`. **Workaround:** None
SDR268388	There are broken links in the API reference for the BlackBerry JDE that are often caused by a difference in capitalization. **Impact:** It is not always possible to follow a link to required information. **Workaround:** None
SDR241224 SDR215580	If you call updateJad.exe and pass two identical filenames as parameters, the application does not correctly update the file. **Impact:** The updateJad.exe can corrupt your application file. **Workaround:** Do not call updateJad.exe and pass two or more identical filenames as parameters.
SDR215764	The Email Services Simulator does not send or receive emails. **Impact:** You cannot test sending and receiving email by using Email Services Simulator. **Workaround:** None

Sample applications

Item	Description
342728	In the Contact Linking Demo sample application, if you link a contact using the Secondary Link to BlackBerry contact menu item, the contact cannot be unlinked.

Item	Description
	Impact: Cannot unlink a linked contact as expected.
	Workaround: None
342695	In the Contact Linking Demo sample application, if you click App Item 2 from the menu, an `IllegalArgumentException` is thrown.
	Impact: You cannot use the App Item 2 menu item in the Contact Linking Demo sample application.
	Workaround: Avoid using the App Item 2 menu item.
342072	The USB Demo sample application freezes the simulator.
	Impact: You cannot use the USB Demo sample application.
	Workaround: None
340769	The Browser Field sample application throws a `RenderingException` when you start the sample application.
	Impact: You cannot use the Browser Field sample application.
	Workaround: None
339237	The Socket Demo client-side sample application does not receive data from the Socket Demo server-side application.
	Impact: The Socket Demo sample application does not work as expected.
	Workaround: None
338306	The Message List sample application throws an exception when the BlackBerry Smartphone Simulator starts up.
	Impact: If you install the Message List sample application, you cannot start the BlackBerry Smartphone Simulator.
	Workaround: Deactivate the messagelistdemo project before building the samples.jdw workspace and starting the BlackBerry Smartphone Simulator.
225624	The VideoRecorder sample application throws a `ControlledAccess` exception when run on a secure BlackBerry Smartphone Simulator.
	Impact: You cannot use the VideoRecorder sample application on a secure simulator.
	Workaround: None

Item	Description
224724	The Multiplayer Tic-Tac-Toe Demo sample application does not send moves to the other player after the first move. **Impact:** You cannot play the game after the first move by each player. **Workaround:** None
113335	The BluetoothDemo application adds a "Connect to <BlackBerry device name>" menu item the sample application is used on a BlackBerry device that has been previously paired with another BlackBerry device. **Impact:** The BluetoothDemo application does not work with paired BlackBerry devices. **Workaround:** None
SDR369449	In the Video Recording Demo sample application, when you click the Commit Record menu item, the application throws a `NullPointerException`. **Impact:** You cannot commit a recording to memory. **Workaround:** None

Legal notice

PERMITTED BY LAW, ANY IMPLIED WARRANTIES OR CONDITIONS RELATING TO THE DOCUMENTATION TO THE EXTENT THEY CANNOT BE EXCLUDED AS SET OUT ABOVE, BUT CAN BE LIMITED, ARE HEREBY LIMITED TO NINETY (90) DAYS FROM THE DATE YOU FIRST ACQUIRED THE DOCUMENTATION OR THE ITEM THAT IS THE SUBJECT OF THE CLAIM.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW IN YOUR JURISDICTION, IN NO EVENT SHALL RIM BE LIABLE FOR ANY TYPE OF DAMAGES RELATED TO THIS DOCUMENTATION OR ITS USE, OR PERFORMANCE OR NON-PERFORMANCE OF ANY SOFTWARE, HARDWARE, SERVICE, OR ANY THIRD PARTY PRODUCTS AND SERVICES REFERENCED HEREIN INCLUDING WITHOUT LIMITATION ANY OF THE FOLLOWING DAMAGES: DIRECT, CONSEQUENTIAL, EXEMPLARY, INCIDENTAL, INDIRECT, SPECIAL, PUNITIVE, OR AGGRAVATED DAMAGES, DAMAGES FOR LOSS OF PROFITS OR REVENUES, FAILURE TO REALIZE ANY EXPECTED SAVINGS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, LOSS OF BUSINESS OPPORTUNITY, OR CORRUPTION OR LOSS OF DATA, FAILURES TO TRANSMIT OR RECEIVE ANY DATA, PROBLEMS ASSOCIATED WITH ANY APPLICATIONS USED IN CONJUNCTION WITH RIM PRODUCTS OR SERVICES, DOWNTIME COSTS, LOSS OF THE USE OF RIM PRODUCTS OR SERVICES OR ANY PORTION THEREOF OR OF ANY AIRTIME SERVICES, COST OF SUBSTITUTE GOODS, COSTS OF COVER, FACILITIES OR SERVICES, COST OF CAPITAL, OR OTHER SIMILAR PECUNIARY LOSSES, WHETHER OR NOT SUCH DAMAGES WERE FORESEEN OR UNFORESEEN, AND EVEN IF RIM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW IN YOUR JURISDICTION, RIM SHALL HAVE NO OTHER OBLIGATION, DUTY, OR LIABILITY WHATSOEVER IN CONTRACT, TORT, OR OTHERWISE TO YOU INCLUDING ANY LIABILITY FOR NEGLIGENCE OR STRICT LIABILITY.

THE LIMITATIONS, EXCLUSIONS, AND DISCLAIMERS HEREIN SHALL APPLY: (A) IRRESPECTIVE OF THE NATURE OF THE CAUSE OF ACTION, DEMAND, OR ACTION BY YOU INCLUDING BUT NOT LIMITED TO BREACH OF CONTRACT, NEGLIGENCE, TORT, STRICT LIABILITY OR ANY OTHER LEGAL THEORY AND SHALL SURVIVE A FUNDAMENTAL BREACH OR BREACHES OR THE FAILURE OF THE ESSENTIAL PURPOSE OF THIS AGREEMENT OR OF ANY REMEDY CONTAINED HEREIN; AND (B) TO RIM AND ITS AFFILIATED COMPANIES, THEIR SUCCESSORS, ASSIGNS, AGENTS, SUPPLIERS (INCLUDING AIRTIME SERVICE PROVIDERS), AUTHORIZED RIM DISTRIBUTORS (ALSO INCLUDING AIRTIME SERVICE PROVIDERS) AND THEIR RESPECTIVE DIRECTORS, EMPLOYEES, AND INDEPENDENT CONTRACTORS.

IN ADDITION TO THE LIMITATIONS AND EXCLUSIONS SET OUT ABOVE, IN NO EVENT SHALL ANY DIRECTOR, EMPLOYEE, AGENT, DISTRIBUTOR, SUPPLIER, INDEPENDENT CONTRACTOR OF RIM OR ANY AFFILIATES OF RIM HAVE ANY LIABILITY ARISING FROM OR RELATED TO THE DOCUMENTATION.

Prior to subscribing for, installing, or using any Third Party Products and Services, it is your responsibility to ensure that your airtime service provider has agreed to support all of their features. Some airtime service providers might not offer Internet browsing functionality with a subscription to the BlackBerry® Internet Service. Check with your service provider for availability, roaming arrangements, service plans and features. Installation or use of Third Party Products and Services with RIM's products and services may require one or more patent, trademark, copyright, or other licenses in order to avoid infringement or violation of third party rights. You are solely responsible for determining whether to use Third Party Products and Services and if any third party licenses are required to do so. If required you are responsible for acquiring them. You should not install or use Third Party Products and Services until all necessary licenses have been acquired. Any Third Party Products and Services that are provided with RIM's products and services are provided as a convenience to you and are provided "AS IS" with no express or implied conditions, endorsements, guarantees, representations, or warranties of any kind by RIM and RIM assumes no liability whatsoever, in relation

thereto. Your use of Third Party Products and Services shall be governed by and subject to you agreeing to the terms of separate licenses and other agreements applicable thereto with third parties, except to the extent expressly covered by a license or other agreement with RIM.

Certain features outlined in this documentation require a minimum version of BlackBerry® Enterprise Server, BlackBerry® Desktop Software, and/or BlackBerry® Device Software.

The terms of use of any RIM product or service are set out in a separate license or other agreement with RIM applicable thereto. NOTHING IN THIS DOCUMENTATION IS INTENDED TO SUPERSEDE ANY EXPRESS WRITTEN AGREEMENTS OR WARRANTIES PROVIDED BY RIM FOR PORTIONS OF ANY RIM PRODUCT OR SERVICE OTHER THAN THIS DOCUMENTATION.